



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**KERJA PRAKTIK – KI141330**

## **Pembuatan Tabel CRUD KONTRAK KINERJA UP3 ITS**

**Unit Pengelolaan dan Pengendalian Program (UP3) - Institut  
Teknologi Sepuluh Nopember Surabaya**

**Jl. Raya ITS, Surabaya, Jawa Timur 60117**

**Periode: 1 Juli 2020 - 1 Oktober 2020**

**Oleh:**

### **Pembimbing Jurusan**

**Dr. Diana Purwitasari, S.Kom., M.Sc.**

### **Pembimbing Lapangan**

**Ridho Rahman Hariadi, S.Kom., M.Sc.**

**DEPARTEMEN INFORMATIKA**

**Fakultas Teknologi Informasi dan Komunikasi**

**Institut Teknologi Sepuluh Nopember**

**Surabaya 2020**



**KERJA PRAKTIK - KI141330**

**Pembuatan Tabel CRUD KONTRAK KINERJA UP3 ITS**

Unit Pengelolaan dan Pengendalian Program (UP3) – Institut  
Teknologi Sepuluh Nopember Surabaya  
Jl. Raya ITS, Surabaya, Jawa Timur 60117  
Periode: 1 Juli 2020 – 1 Oktober 2020

Oleh:

**Pembimbing Jurusan**

Dr.Diana Purwitasari , S.Kom., M.Sc.

**Pembimbing Lapangan**

Ridho Rahman Hariadi, S.Kom., M.Sc.

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember  
Surabaya 2020

*[Halaman ini sengaja dikosongkan]*

## LEMBAR PENGESAHAN

### KERJA PRAKTIK

#### **Pembuatan Tabel CRUD KONTRAK KINERJA UP3 ITS**

Oleh:

Kevin Ashil Faadilah

05111740000178

Mengetahui,

Institut Teknologi Sepuluh Nopember

Pembimbing Kerja Praktik

Menyetujui,

Dosen Pembimbing

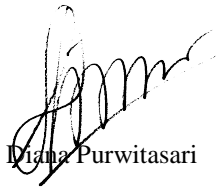
Kerja Praktik



Ridho Rahman Hariadi

S.Kom., M.Sc

NIP. 198702132014041001



Dr. Diana Purwitasari

S.Kom.,M.Sc.,

NIP. 197804102003122001

SURABAYA

14 Januari, 2021

*[Halaman ini sengaja dikosongkan]*

*[Halaman ini sengaja dikosongkan]*

## **Pembuatan Tabel CRUD KONTRAK KINERJA UP3ITS**

**Nama Mahasiswa : Kevin Ashil Faadilah**  
**NRP : 05111740000178**

**Departemen : Teknik Informatika FTEIC-ITS**  
**Pembimbing Jurusan : Dr. Diana Purwitasari S.Kom.,M.Sc.**  
**Pembimbing Lapangan : Ridho Rahman Hariadi S.Kom., M.Sc.**

## **ABSTRAK**

Semakin besar sebuah organisasi, semakin banyak pula data yang harus dikelola. Tidak terkecuali Institut Teknologi Sepuluh Nopember. Oleh karena itu Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya sedang mengembangkan sebuah sistem input untuk data kinerja berbasis web untuk membantu menyelesaikan persoalan tersebut.

Dalam kerja praktek ini kami ditugaskan oleh Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya untuk menambahkan beberapa fitur yang dibutuhkan seperti query Create, query Read, query Update, query Delete, dan menambah trigger. Sistem ini dibangun menggunakan framework Vue.js dengan database SQL Server beserta library JExcel.

***Kata kunci: Query, database, Vue.js, JExcel, S QL Server***



## **KATA PENGANTAR**

Puji syukur kami haturkan kepada Allah SWT karena berkat rahmat-Nya kami dapat melaksanakan salah satu kewajiban kami sebagai mahasiswa Departemen Informatika, yakni Kerja Praktek (KP).

Kami menyadari masih ada kekurangan baik dalam pelaksanaan kerja praktik maupun penyusunan buku laporan ini. Namun, kami berharap buku laporan ini dapat menambah wawasan pembaca dan dapat menjadi sumber referensi. Kami mengharapkan kritik dan saran yang membangun untuk kesempurnaan buku laporan kerja praktik ini.

Melalui buku ini, kami juga ingin menyampaikan rasa terima kasih kepada orang-orang yang telah membantu, baik langsung maupun tidak langsung, dalam pelaksanaan kerja praktik hingga penyusunan laporan. Orang-orang tersebut antara lain adalah:

1. Kedua orang tua penulis.
2. Ibu Diana Purwitasari S.Kom., M.Sc., selaku dosen pembimbing kerja praktik.
3. Bapak Ary Mazharuddin Shiddiqi S.Kom., M.Comp., Ph.D, selaku koordinator Kerja Praktik.
4. Bapak Ridho Rahman Hariadi, S.Kom., M.Sc., selaku pembimbing lapangan kami di Unit Pengelolaan dan Pengendalian Program (UP3) ITS.

Surabaya, 14 Januari 2021  
Kevin Ashil Faadilah

*[Halaman ini sengaja dikosongkan]*

## **DAFTAR ISI**

LEMBAR PENGESAHAN	4
ABSTRAK	8
KATA PENGANTAR	9
DAFTAR ISI	11
BAB I	18
PENDAHULUAN	18
1.1 Latar Belakang	18
1.2 Tujuan	18
1.3 Manfaat	19
1.4 Rumusan Masalah	19
1.5 Lokasi dan Waktu Kerja Praktik	20
1.6 Metodologi Kerja Praktik	20
1.7 Sistematika Laporan	22
BAB II	25
PROFIL PERUSAHAAN	25
2.1 Profil Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya	25
2.1.1 Tugas Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya	25

2.1.2 Fungsi Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya	25
BAB III	26
TINJAUAN PUSTAKA	26
3.1 Pemrograman Database	26
3.2 Visual Studio Code	26
3.3 Vue.js	26
3.4 JExcel	27
3.5 Postman	27
3.6 SQL Server Management Studio(SSMS)	27
BAB IV	28
IMPLEMENTASI SISTEM	28
4.1. Query Create	28
4.1.1 Deskripsi	28
4.1.2. Parameter	28
4.1.3. Data	28
4.1.4. Tabel Query	28
4.1.5. Source Code	32
4.2. Query Read	33
4.2.1 Deskripsi	33

4.2.2. Parameter	33
4.2.3. Data	33
4.2.4. Tabel Query	33
4.2.5. Source Code	35
4.3. Query Update	37
4.3.1. Deskripsi	37
4.3.2. Parameter	37
4.3.3. Data	37
4.3.4. Tabel Query	39
4.3.5. Source Code	40
4.4. Query Delete	41
4.4.1. Deskripsi	41
4.4.2. Parameter	41
4.4.3. Data	41
4.4.4. Tabel Query	42
4.4.4. Source Code	43
4.5. Trigger	44
4.5.1. Deskripsi	44
4.5.2. Parameter	44
4.5.3. Data	44

4.5.4. Source Code	44
4.6. Routing API	45
4.5.1. Deskripsi	45
4.5.2. Parameter	45
4.5.3. Data	45
4.5.4. Source Code	45
BAB V	46
PENGUJIAN DAN EVALUASI	46
5.1 Tujuan Pengujian	46
5.2 Kriteria Pengujian	46
5.3 Skenario Pengujian	46
5.4 Hasil Pengujian	46
5.5 Evaluasi Pengujian	47
BAB VI	51
KESIMPULAN DAN SARAN	51
6.1 Kesimpulan	51
6.2 Saran	51
DAFTAR PUSTAKA	53
BIODATA PENULIS	55

*[Halaman ini sengaja dikosongkan]*

## DAFTAR TABEL

Tabel	4.1	Detail	Query	Create	Tabel	
d_000_dosen_inbound						30
Tabel	4.2	Detail	Query	Create	Tabel	
d_000_dosen_outbound						31
Tabel4.52	Detail	Query	Read	Tabel d_000_dosen_inbound		34
Tabel 4.53	Detail	Query	Read	Tabel d_000_dosen_outbound		35
Tabel 4.112	Detail	Query	Update	Tabel d_000_publikasi_scopus_scholar		38
Tabel 4.113	Detail	Query	Update	Tabel d_037_bukuIsbn		39
Tabel 4.161	Detail	Query	Delete	Tabel d_000_dosen_inbound		44
Tabel 4.162	Detail	Query	Delete	Tabel d_000_dosen_outbound		45



## **DAFTAR GAMBAR**

Gambar 5.1 Uji Query Create dengan postman	47
Gambar 5.2 Uji Query Create dengan front end	47
Gambar 5.3 Uji Query Read dengan postman	47
Gambar 5.4 Uji Query Read dengan front end	48
Gambar 5.5 Uji Query Update dengan postman	48
Gambar 5.6 Uji Query Update dengan front end	48
Gambar 5.7 Query Delete dengan postman	49
Gambar 5.8 Query Delete dengan front end	49
Gambar 5.9 Trigger dengan DBMS	49

## **DAFTAR KODE**

Kode 4.1 Source Code Query Create	32
Kode 4.2 Source Code Query Read dengan filter satuan kerja	32
Kode 4.3 Source Code Query Update	36
Kode 4.4 Source Code Query Delete	43
Kode 4.5 Source Code Trigger	44
Kode 4.6 Source Code Routing API	45

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Tahun 2019, ITS menduduki peringkat 4 Perguruan Tinggi terbaik di Indonesia dan dinyatakan sebagai Perguruan Tinggi Negeri Berbadan Hukum (PTN-BH) terbaik (peringkat 1) dalam Pelaksanaan Program, Kegiatan, dan Anggaran. ITS juga menerima Penghargaan Widyapadi (peringkat 3) sebagai Perguruan Tinggi terinovasi dan Penghargaan SINTA untuk kategori publikasi terbanyak. Di level internasional, ITS menduduki peringkat ke-3 terbaik Perguruan Tinggi di Indonesia versi Times Higher Education (THE) World University Ranking tahun 2019 dan 2020. Sedangkan, di level Asia Pasifik, ITS menduduki peringkat 201+. ITS juga memiliki banyak organisasi di bawahnya yang dimana mendorong mereka untuk mendigitalisasikan dirinya. Dengan semangat untuk terus berbenah dilakukan oleh ITS<sup>[7]</sup>. Untuk mewujudkan semangat tersebut dengan melakukan sinergi antar organisasi yang berada dalam naungan ITS Surabaya. Salah satunya adalah organisasi UP3 (Unit Pengelolaan dan Pengembangan Program) Pada saat ini sedang melakukan pembenahan untuk bagian internal dalam penginputan data.

Oleh karena itu melalui pada kesempatan kerja praktik kali ini, Pihak UP3 memberi kesempatan pada penulis membantu beberapa pekerjaan, utamanya dalam membuat query untuk melakukan CRUD (Create, Read, Update, Delete) pada form web KONTRAK KINERJA UP3 ITS dan membuat trigger untuk database pada kerja praktek yang kami laksanakan.

### **1.2 Tujuan**

Tujuan kerja praktik ini adalah untuk menyelesaikan kewajiban kuliah kerja praktik di Institut Teknologi Sepuluh Nopember memiliki jumlah SKS sebanyak dua. Selain itu juga untuk membuat aplikasi berbasis web untuk keperluan input data kinerja

bagi para karyawan di Unit Pengelolaan dan Pengendalian Program (UP3) ITS dan civitas akademik Institut Teknologi Sepuluh Nopember.

### **1.3 Manfaat**

Manfaat dari pembuatan website yang kami buat sama seperti tujuan kami yaitu untuk membuat aplikasi berbasis web untuk membantu para karyawan dan civitas akademik dalam melakukan input data kinerja. Kemudian untuk penulis sebagai pengaplikasian ilmu yang sudah dimiliki dan menambah pengalaman terhadap dunia kerja.

### **1.4 Rumusan Masalah**

Berikut ini rumusan masalah pada kerja praktik pembuatan website kinerja :

1. Bagaimana sebuah aplikasi berbasis web mengolah data input utamanya untuk query?

### **1.5 Lokasi dan Waktu Kerja Praktik**

Dikarenakan adanya pandemi saat kerja praktik ini dilaksanakan, detail lokasi dan waktu pengerjaan kerja praktik adalah sebagai berikut:

Lokasi : Work From Home (WFH)  
Alamat : Jln. Rungkut Harapan blok G-01 Surabaya

Waktu : 1 Juli 2020 - 1 Oktober 2020  
Hari Kerja : Senin - Jumat  
Jam Kerja : 08.00 WIB – 17.00 WIB

### **1.6 Metodologi Kerja Praktik**

Tahapan pengerjaan kerja praktik dapat dijabarkan sebagai berikut:

## **1.Perumusan Masalah**

Untuk mengetahui permasalahan apa yang harus diselesaikan, diberikan penjelasan mengenai alasan mengapa fungsi-fungsi tersebut dibutuhkan. Dijelaskan juga secara rinci mengenai bagaimana alur sistem itu akan berjalan. penjelasan mengenai hal ini dijelaskan oleh pembimbing lapangan kerja praktik. Dari penjelasannya dihasilkan catatan-catatan penting mengenai gambaran sistem berbasis web yang akan dibuatkan fungsi dan query untuk mempermudah kinerjanya. Dengan begitu dapat diputuskan bahasa pemrograman database yang digunakan adalah SQL, website dibangun menggunakan framework Vue.JS, dengan tools pendukung SQL Server Management Studio (SSMS), postman dan Visual Studio Code.

## **2.Studi Literatur**

Setelah ditentukan *database* , bahasa pemrograman, dan serta *tools* tambahan yang akan digunakan, dilakukan studi literatur mengenai cara implementasinya dalam membuat fungsi query.

Pada tahap ini dilakukan proses pencarian, pembelajaran, pengumpulan dan pemahaman informasi serta literatur yang berkaitan untuk membantu dalam implementasi fungsi query ini. Informasi bisa didapat dari internet untuk istilah-istilah umum yang digunakan dalam pengimplementasian suatu fungsi query.

## **3.Analisis dan Perancangan**

Tahap ini meliputi penjelasan tentang fungsi query berdasarkan studi literatur dan pembelajaran pembuatan fungsi query yang sudah ada. Tahap ini mendefinisikan alur dari implementasi.

Langkah-langkah yang di kerjakan juga didefinisikan ditahap ini sebagai berikut:

- a. Memahami elemen dan isi dari seluruh tabel yang ada di dalam database

- b. Memahami primary key dan foreign key yang ada pada tabel
- c. Memahami trigger yang ada pada tabel
- d. Memahami tools SQL Server Management Studio sebagai tools pendukung yang digunakan dalam membuat fungsi query

#### **4.Implementasi Sistem**

Implementasi sistem didasarkan karena adanya kebutuhan dari full-stack developer untuk mempermudah jalannya suatu web dengan database yang sudah ter query. Pada tahap ini kami membuat beberapa fungsi query yang dibutuhkan untuk melakukan input, delete, select, dan update data. Kami juga melakukan beberapa perubahan pada database yang sudah ada. Pengerjaan dilakukan selama kurang lebih tiga bulan.

#### **5.Pengujian dan Evaluasi**

Pengujian dilakukan dengan menguji query yang telah dibuat. Tahap awal setelah query jadi dilakukan testing secara lokal dan didemokan kepada pembimbing lapangan kerja praktik apakah query yang dibuat sudah sesuai dengan kebutuhan. Dengan ini akan dihasilkan hasil evaluasi apakah fungsi sudah sesuai dengan tujuan dan kebutuhan atau belum.

#### **6.Kesimpulan dan Saran**

Kesimpulan yang kami dapatkan diantaranya adalah perlunya penyesuaian atau pemahaman terhadap masalah yang dibutuhkan untuk membuat suatu fungsi dengan tingkat kesulitan yang berbeda-beda dan dibutuhkan komunikasi jika ada sesuatu yang kurang jelas dengan permasalahan yang diberikan agar pada saat pengerjaan tidak terjadi kesalahan dalam pemecahan masalah, karena akan berpengaruh terhadap fungsi yang dibuat terlebih fungsi yang dibuat merupakan

fungsi yang dapat menghapus atau merubah data dari database client.

## **1.7 Sistematika Laporan**

Laporan kerja praktik ini terdiri dari tujuh bab dengan rincian sebagai berikut:

### **1.Bab I Pendahuluan**

Pada bab ini dijelaskan tentang latar belakang permasalahan, tujuan, waktu pelaksanaan, serta sistematika pengerjaan kerja praktik dan juga penulisan laporan kerja praktik.

### **2.Bab II Profil Perusahaan**

Pada bab ini, dijelaskan secara rinci tentang profil perusahaan tempat kami melaksanakan kerja praktik, yakni Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya.

### **3.Bab III Tinjauan Pustaka**

Pada bab ini, dijelaskan mengenai tinjauan pustaka dan literatur yang digunakan dalam penyelesaian kerja praktik di Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya.

### **4.Bab IV Implementasi Sistem**

Pada bab ini, berisi penjelasan tahap-tahap yang dilakukan untuk proses implementasi fungsi query dan trigger dengan Bahasa pemrograman database SQL.

### **5.Bab V Pengujian dan Evaluasi**

Pada bab ini, dijelaskan tentang hasil pengujian dan evaluasi dari sistem yang telah dikembangkan selama pelaksanaan kerja praktik di Unit Pengelolaan dan

Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya.

## **6.Bab VI Kesimpulan dan Saran**

Pada bab ini, dipaparkan kesimpulan yang dapat diambil dan juga saran selama pengerjaan kerja praktik.



## **BAB II**

### **PROFIL PERUSAHAAN**

#### **2.1 Profil Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya**

Unit Pengelolaan dan Pengendalian Program (UP3) Institut Teknologi Sepuluh Nopember (ITS) Surabaya adalah sebuah unit yang dibentuk oleh ITS yang sesuai namanya dibentuk untuk mengelola dan mengendalikan program-program yang dijalankan oleh ITS.

##### **2.1.1 Tugas Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya**

- Membantu merumuskan kebijakan dalam subbidang pengelolaan dan pengendalian program
- Membina, mengarahkan, dan mengembangkan kompetensi dan talenta personil dalam unit kerja
- Menyelenggarakan program kerja dan kegiatan yang selaras dengan kebijakan dalam subbidang pengelolaan dan pengendalian program
- Mengevaluasi kinerja hasil program kerja dan kegiatan dalam subbidang pengelolaan dan pengendalian program
- Menyelenggarakan pelayanan prima dalam subbidang pengelolaan dan pengendalian program dengan prinsip reformasi birokrasi dan zona integritas

##### **2.1.2 Fungsi Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya**

- Penyiapan penyusunan rencana induk pengembangan, rencana strategis organisasi, dan rencana operasional organisasi
- Pengordinasian penyusunan rencana kegiatan dan anggaran
- Pengelolaan, pengendalian, dan pengawasan program kerja

- Penyelarasan program kerja ITS dengan seluruh unit di bawah Rektor
- Pengoordinasian perancangan pangkalan data terpadu ITS
- Pengelolaan data stewardship dari berbagai sumber secara terpadu yang diperlukan untuk pengambilan kebijakan ITS dan pelaporan
- Pengukuran dan pelaporan kinerja organisasi untuk semua level organisasi
- Pengoordinasian dan penyusunan pelaporan organisasi secara berkala
- Penyelenggaraan sistem pengendalian intern ITS
- Perancangan dan pemantauan strategi organisasi terkait pemeringkatan ITS di level nasional
- Penyelenggaraan survei kepuasan stakeholder terhadap layanan organisasi
- Pelaksanaan identifikasi dan penelusuran hambatan dan penyimpangan pelaksanaan program kerja ITS, serta memberikan alternatif usulan-usulan perbaikan
- Pelaksanaan evaluasi dan pelaporan penyelenggaraan program kerja subbidang pengelolaan dan pengendalian program

## **BAB III**

### **TINJAUAN PUSTAKA**

Pada bab ini, akan dijelaskan mengenai dasar teori yang digunakan selama proses kerja praktik.

#### **3.1 Pemrograman Database**

Basis data (database) adalah sekelompok data yang disimpan dan diatur sedemikian rupa sehingga bisa dikelola secara efisien. Untuk mengatur, menyimpan, dan memodifikasi data yang berada di dalam sebuah database digunakan perangkat lunak (software) yang disebut dengan DBMS.

SQL(Structured Query Language) adalah bahasa yang digunakan untuk mengakses dan memanipulasi database.

#### **3.2 Visual Studio Code**

Visual Studio Code adalah editor source code yang tersedia untuk Windows, macOS, dan Linux. Program ini tersedia dengan built-in support untuk JavaScript, TypeScript dan Node.js dan memiliki ekosistem yang melimpah akan ekstensi untuk bahasa pemrograman lain (seperti C++, C#, Java, Python, PHP, Go).

#### **3.3 Vue.js**

Vue.js adalah framework progresif untuk membangun user interface. Framework ini bisa diintegrasikan dengan berbagai library. Vue.js sendiri didistribusikan secara open source.

#### **3.4 JExcel**

JExcel adalah sebuah plugin javascript yang berfungsi untuk membuat spreadsheet interaktif berbasis web yang kompatibel dengan berbagai macam software spreadsheet.

### **3.5 Postman**

Postman adalah API client yang memudahkan developer untuk membuat, menguji, mendistribusikan, dan mendokumentasikan API.

### **3.6 SQL Server Management Studio(SSMS)**

SSMS adalah sebuah software yang berfungsi untuk mengelola infrastruktur SQL. SSMS menyediakan tools untuk mengkonfigurasi, mengelola, dan mengawasi server dan database SQL.

## **BAB IV**

### **IMPLEMENTASI SISTEM**

Pada bab ini menjelaskan tahap implementasi yang dilakukan. Implementasi akan dibagi menjadi beberapa bagian, yaitu bagian query Create, query Read, query Update, query Delete, trigger, dan routing API.

#### **4.1. Query Create**

##### **4.1.1 Deskripsi**

Membuat query Create untuk menginput data kosong kecuali id kedalam sistem.

##### **4.1.2. Parameter**

Tidak ada parameter yang digunakan.

##### **4.1.3. Data**

Untuk query Create pada form JExcel data yang terinput berupa data kosong kecuali id sebagai/primary key.

##### **4.1.4. Tabel Query**

*Tabel 4.1 Detail Query Create Tabel d\_000\_dosen\_inbound*

Route	/v_d_000_dosen_inbound
Query	<pre> insert into KINERJA_PENDUKUNG_DEV.kinerja.d_000_d osen_inbound(unit_departemen,satker_kode,status, title,nama_dosen,nama_program,jenis_program,un iversitas_mitra,negara,pendanaan,tgl_mulai,tgl_sel esai,durasi,satuan_durasi,tahun_program) VALUES(@unit_departemen,@satker_kode,@stat us,@title,@nama_dosen,@nama_program,@jenis _program,@universitas_mitra,@negara,@pendana an,@tgl_mulai,@tgl_selesai,@durasi,@satuan_dur asi,@tahun_program) </pre>
Parameter	-
Return	-

*Tabel 4.2 Detail Query Create Tabel d\_000\_dosen\_outbound*

Route	v_d_000_dosen_outbound
Query	<pre> insert into KINERJA_PENDUKUNG_DEV.kinerja.d_000_d osen_outbound(departemen,satker_kode,status,title ,nama_dosen,nip_dosen,nama_program,jenis_prog ram,universitas_mitra,negara,pendanaan,tgl_mulai, tgl_selesai,durasi,satuan_durasi,tahun_program) VALUES( @departemen,@satker_kode,@status,@ title,@nama_dosen,@nip_dosen,@nama_program, @jenis_program,@universitas_mitra,@negara,@p endanaan,@tgl_mulai,@tgl_selesai,@durasi,@sat uan_durasi,@tahun_program) </pre>
Parameter	-
Return	-

## 4.1.5. Source Code

Source Code untuk query Create dapat dilihat pada kode 4.1.

```
41 // { name: 'tahun_program', sqltype: sql.Int, value: req.body.tahun_program }
42 // }
43 // var query = "insert into KINERJA_PENDUKUNG_DEV.dbo.d_000_mahasiswa_outbound(satker_kode,nama_mahasiswa,nrp,negara_tujuan,nama_program,tgl_mulai,tgl_selesai,verifikasi,bu
44 // nding)values(?,?,?,?,?,?,?)";
45 // });
46 router.route('/v_d_000_dosen_inbound').post(function(req, res){
47     var params = [
48         { name: 'unit_departemen', sqltype: sql.VarChar(255), value: req.body.unit_departemen },
49         { name: 'satker_kode', sqltype: sql.VarChar(20), value: req.body.satker_kode },
50         { name: 'status', sqltype: sql.VarChar(64), value: req.body.status },
51         { name: 'title', sqltype: sql.VarChar(64), value: req.body.title },
52         { name: 'nama_dosen', sqltype: sql.VarChar(255), value: req.body.nama_dosen },
53         { name: 'nama_program', sqltype: sql.VarChar, value: req.body.nama_program },
54         { name: 'jenis_program', sqltype: sql.VarChar, value: req.body.jenis_program },
55         { name: 'universitas_nitra', sqltype: sql.VarChar, value: req.body.universitas_nitra },
56         { name: 'negara', sqltype: sql.VarChar(64), value: req.body.negara },
57         { name: 'pendanaan', sqltype: sql.VarChar, value: req.body.pendanaan },
58         { name: 'tgl_mulai', sqltype: sql.Date, value: req.body.tgl_mulai },
59         { name: 'tgl_selesai', sqltype: sql.Date, value: req.body.tgl_selesai },
60         { name: 'durasi', sqltype: sql.Int, value: req.body.durasi },
61         { name: 'satuan_durasi', sqltype: sql.VarChar(64), value: req.body.satuan_durasi },
62         { name: 'tahun_program', sqltype: sql.Int, value: req.body.tahun_program }
63     ];
64     var query = "insert into KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_inbound(unit_departemen,satker_kode,status,title,nama_dosen,nama_program,jenis_program,universitas_nitra,
65     executeQuery(res, query, params)
66     });
67 router.route('/v_d_000_dosen_outbound').post(function(req, res){
68     var params = [
69         { name: 'departemen', sqltype: sql.VarChar(255), value: req.body.departemen },
70         { name: 'satker_kode', sqltype: sql.VarChar(20), value: req.body.satker_kode },
71         { name: 'status', sqltype: sql.VarChar(64), value: req.body.status },
72         { name: 'title', sqltype: sql.VarChar(64), value: req.body.title },
73         { name: 'nama_dosen', sqltype: sql.VarChar(255), value: req.body.nama_dosen },
74         { name: 'nama_program', sqltype: sql.VarChar(255), value: req.body.nama_program },
75         { name: 'jenis_program', sqltype: sql.VarChar, value: req.body.jenis_program },
76         { name: 'universitas_nitra', sqltype: sql.VarChar, value: req.body.universitas_nitra },
77         { name: 'negara', sqltype: sql.VarChar(64), value: req.body.negara },
78         { name: 'pendanaan', sqltype: sql.VarChar, value: req.body.pendanaan },
79         { name: 'tgl_mulai', sqltype: sql.Date, value: req.body.tgl_mulai },
80         { name: 'tgl_selesai', sqltype: sql.Date, value: req.body.tgl_selesai },
81         { name: 'durasi', sqltype: sql.Int, value: req.body.durasi },
82         { name: 'satuan_durasi', sqltype: sql.VarChar(64), value: req.body.satuan_durasi },
83         { name: 'tahun_program', sqltype: sql.Int, value: req.body.tahun_program }
84     ];
85     var query = "insert into KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_outbound(unit_departemen,satker_kode,status,title,nama_dosen,nama_program,jenis_program,universitas_nitra,
86     executeQuery(res, query, params)
87     });
```



## 4.2. Query Read

### 4.2.1 Deskripsi

Membuat query Read untuk menampilkan data yang relevan dengan form dan untuk dropdown.

### 4.2.2. Parameter

Parameter berupa kolom-kolom yang sesuai untuk masing-masing form.

### 4.2.3. Data

Data yang digunakan adalah satker (satuan kerja) tabel yang sesuai dengan masing-masing form.

### 4.2.4. Tabel Query

*Tabel 4.52 Detail Query Read Tabel d\_000\_dosen\_inbound/:satker*

Route	v_d_000_dosen_inbound/:satker
Query jika satuan kerja = 1	select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_inbound
Query jika satuan kerja!= 1	select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_inbound where satker_kode like @satker + '%'
Parameter	{ name: 'satker', sqltype: sql.VarChar, value: satker }

Return	-
--------	---

*Tabel 4.53 Detail Query Read Tabel d\_000\_dosen\_outbound*

Route	v_d_000_dosen_outbound/:satker
Query jika satuan kerja = 1	select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_outbound
Query jika satuan kerja!= 1	select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_outbound where satker_kode like @satker + '%'
Parameter	{ name: 'satker', sqltype: sql.VarChar, value: satker }
Return	-

#### **4.2.5. Source Code**

Source Code yang digunakan untuk query Read ditampilkan pada kode 4.2.

```

1 var express = require('express');
2 var router = express.Router();
3 var sql = require('mysql');
4 var executeQuery = require('../Connection');
5 const e = require('express');
6
7 // :satker = kode satker akun yang login
8
9 var routes = function(){
10   router.route('/v_d_000_dosen_inbound/:satker').get(function(req, res) {           // Akses ??? | Read Only
11     var satker = req.params.satker
12     if (satker == 1) { // Role Admin
13       var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_inbound ";
14       executeQuery(res, query)
15     } else { // Role User
16       var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_inbound where satker_kode like @satker + '%' ";
17       var params = [
18         { name: 'satker', sqldata: sql.VarChar, value: satker }
19       ]
20       executeQuery(res, query, params);
21     }
22   });
23   router.route('/v_d_000_dosen_outbound/:satker').get(function(req, res) {           // Akses ?? | read Only
24     var satker = req.params.satker
25     if (satker == 1) {
26       var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_outbound ";
27       executeQuery(res, query)
28     } else {
29       var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_outbound where satker_kode like @satker + '%' ";
30       var params = [
31         { name: 'satker', sqldata: sql.VarChar, value: satker }
32       ]
33       executeQuery(res, query, params);
34     }
35   });
36   router.route('/v_d_000_mahasiswa_inbound/:satker').get(function(req, res) {
37     // var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_mahasiswa_inbound WHERE satker_departemen = @satker" +
38     // " or nama_departemen = @satker or nama_fakultas = @satker"
39     // var params = [
40     //   { name: 'satker', sqldata: sql.VarChar, value: satker }
41     // ];
42     // executeQuery(res, query, params);
43     // });
44     // router.route('/v_d_000_mahasiswa_outbound/:satker').get(function(req, res) {
45     //   var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_mahasiswa_outbound WHERE departemen = @satker"
46     //   var params = [
47     //     { name: 'satker', sqldata: sql.VarChar, value: satker }
48     //   ]
49     //   executeQuery(res, query, params);
50     //   });
51     // executeQuery(res, query, params);
52     // });
53     router.route('/v_d_000_publikasi_scopus_scholar/:satker').get(function(req, res) {           // Wanna check if this exist in db but ugh
54       var satker = req.params.satker
55       if (satker == 1) {
56         var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_publikasi_scopus_scholar";
57         executeQuery(res, query)
58       } else {
59         var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_000_publikasi_scopus_scholar where satker_kode like @satker + '%'";
60         var params = [
61           { name: 'satker', sqldata: sql.VarChar, value: satker }
62         ]
63         executeQuery(res, query, params);
64       }
65     }
66   });
67   router.route('/v_d_037_bukuIsbn/:satker').get(function(req, res) {           // Akses Departemen
68     var satker = req.params.satker
69     if (satker == 1) { // Role Admin
70       var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_037_bukuIsbn";
71       executeQuery(res, query);
72     } else { // Role User
73       var query = "select * from KINERJA_PENDUKUNG_DEV.kinerja.d_037_bukuIsbn where satker_kode like @satker + '%' ";
74       var params = [
75         { name: 'satker', sqldata: sql.VarChar, value: satker }
76       ]
77       executeQuery(res, query, params);
78     }
79   });
80   router.route('/v_d_072_sertifikasiDosen/:satker').get(function(req, res) {           // Access Level Depptemen

```

### **4.3. Query Update**

#### **4.3.1. Deskripsi**

Membuat query Update untuk mengisi bagian form yang kosong atau mengupdate bagian form yang sudah terisi dengan data yang ingin di input oleh user.

#### **4.3.2. Parameter**

Parameter berupa id.

#### **4.3.3. Data**

Data yang digunakan untuk query Update adalah data yang ingin di input oleh user.

#### **4.3.4. Tabel Query**

*Tabel 4.112 Detail Query Update Tabel d\_000\_publikasi\_scopus\_scholar*

Route	v_d_000_publikasi_scopus_scholar/:id
Query	<p>Update  KINERJA_PENDUKUNG_DEV.kinerja.d_000_p  ublikasi_scopus_scholar SET  satker_kode=@satker_kode,nama_lab=@nama_la  b,id_author_scopus=@id_author_scopus,id_author  _google=@id_author_google,title=@title,publicati  onName=@publicationName,tahun=@tahun,jenis  _publicasi=@jenis_publicasi,verifikasi=@verifika  si,bukti=@bukti,  keterangan_verifikasi=@keterangan_verifikasi  where id=@id</p>
Parameter	{ name: 'id', sqltype: sql.BigInt, value: req.params.id },
Return	-

*Tabel 4.113 Detail Query Update Tabel d\_037\_bukuIsbn*

Route	v_d_037_bukuIsbn/:id
Query	<p>update  KINERJA_PENDUKUNG_DEV.kinerja.d_037_bukuIsbn SET  satker_kode=@satker_kode,departemen=@departemen,id_buku_isbn=@id_buku_isbn,judul_buku=@judul_buku,tahun=@tahun,penerbit=@penerbit,tempat=@tempat,ISBN=@ISBN,author_name=@author_name,sinta_id=@sinta_id,nip_dosen1=@nip_dosen1,nama_dosen1=@nama_dosen1,nip_dosen2=@nip_dosen2,nama_dosen2=@nama_dosen2,nip_dosen3=@nip_dosen3,nama_dosen3=@nama_dosen3,nip_dosen4=@nip_dosen4,nama_dosen4=@nama_dosen4,nip_dosen5=@nip_dosen5,nama_dosen5=@nama_dosen5,verifikasi=@verifikasi,bukti=@bukti,satker_kode_pengisi=@satker_kode_pengisi,added_by=@added_by,keterangan_verifikasi=@keterangan_verifikasi  where id=@id</p>
Parameter	{ name: 'id', sqltype: sql.Int, value: req.params.id },
Return	-

## 4.3.5. Source Code

Source Code untuk query Update dapat dilihat pada kode

4.3.

```
81 // (name: 'pendaftaran', saltype: sql.VarChar, value: req.body.pendaftaran ),
82 // (name: 'tgl_mulai', saltype: sql.Date, value: req.body.tgl_mulai ),
83 // (name: 'tgl_selesai', saltype: sql.Date, value: req.body.tgl_selesai ),
84 // (name: 'duraail', saltype: sql.Int, value: req.body.duraail ),
85 // (name: 'status_duraail', saltype: sql.VarChar(4), value: req.body.status_duraail ),
86 // (name: 'tahun_program', saltype: sql.Int, value: req.body.tahun_program )
87 // }
88 // var query = "UPDATE KINERJA_PENINGKAT_DEV.kinerja_d_000_dosen_outbound SET departemen@departemen,satker_kode@satker_kode,status@status,title@title,nama_dosen@nama_d
89 // executeQuery(res, query, params)
90 // }
91 router.route('/v_d_000_publikasi_scopus_scholar/id').put(function(req, res){
92   var params = [
93     (name: 'id', saltype: sql.BigInt, value: req.params.id ),
94     (name: 'satker_kode', saltype: sql.VarChar(20), value: req.body.satker_kode ),
95     (name: 'nama_lab', saltype: sql.VarChar(50), value: req.body.laboratorium ),
96     (name: 'id_author_scopus', saltype: sql.BigInt, value: req.body.id_author_scopus ),
97     (name: 'id_author_google', saltype: sql.BigInt, value: req.body.id_author_google ),
98     (name: 'title', saltype: sql.VarChar, value: req.body.title ),
99     (name: 'publicationname', saltype: sql.VarChar, value: req.body.publicationName ),
100    (name: 'tahun', saltype: sql.Int, value: req.body.tahun ),
101    (name: 'jenis_publikasi', saltype: sql.VarChar(50), value: req.body.jenis_publikasi ),
102    (name: 'verifikasi', saltype: sql.Int, value: req.body.verifikasi ),
103    (name: 'bukti', saltype: sql.VarChar(500), value: req.body.bukti ),
104    (name: 'keterangan_verifikasi', saltype: sql.VarChar, value: req.body.keterangan_verifikasi )
105  ];
106
107   var query = "UPDATE KINERJA_PENINGKAT_DEV.kinerja_d_000_publikasi_scopus_scholar SET satker_kode@satker_kode,nama_lab@nama_lab,id_author_scopus@id_author_scopus,id_author
108   executeQuery(res, query, params)
109
110 });
111 router.route('/v_d_037_bukuisbn/id').put(function(req, res){
112   var params = [
113     (name: 'id', saltype: sql.Int, value: req.params.id ),
114     (name: 'satker_kode', saltype: sql.VarChar(20), value: req.body.satker_kode ),
115     (name: 'departemen', saltype: sql.VarChar(50), value: req.body.departemen ),
116     (name: 'id_buku_isbn', saltype: sql.BigInt, value: req.body.id_buku_isbn ),
117     (name: 'judul_buku', saltype: sql.VarChar, value: req.body.judul_buku ),
118     (name: 'tahun', saltype: sql.VarChar, value: req.body.tahun ),
119     (name: 'penerbit', saltype: sql.VarChar, value: req.body.penerbit ),
120
121     (name: 'tema', saltype: sql.VarChar, value: req.body.tema ),
122     (name: 'ISBN', saltype: sql.VarChar, value: req.body.ISBN ),
123     (name: 'author_name', saltype: sql.VarChar, value: req.body.author_name ),
124     (name: 'dieta_id', saltype: sql.BigInt, value: req.body.dieta_id ),
125     (name: 'nip_dosen1', saltype: sql.VarChar(50), value: req.body.nip_dosen1 ),
126     (name: 'nama_dosen1', saltype: sql.VarChar(50), value: req.body.nama_dosen1 ),
127     (name: 'nip_dosen2', saltype: sql.VarChar(50), value: req.body.nip_dosen2 ),
128     (name: 'nama_dosen2', saltype: sql.VarChar(50), value: req.body.nama_dosen2 ),
129     (name: 'nip_dosen3', saltype: sql.VarChar(50), value: req.body.nip_dosen3 ),
130     (name: 'nama_dosen3', saltype: sql.VarChar(50), value: req.body.nama_dosen3 ),
131     (name: 'nip_dosen4', saltype: sql.VarChar(50), value: req.body.nip_dosen4 ),
132     (name: 'nama_dosen4', saltype: sql.VarChar(50), value: req.body.nama_dosen4 ),
133     (name: 'nip_dosen5', saltype: sql.VarChar(50), value: req.body.nip_dosen5 ),
134     (name: 'nama_dosen5', saltype: sql.VarChar(50), value: req.body.nama_dosen5 ),
135     (name: 'verifikasi', saltype: sql.Int, value: req.body.verifikasi ),
136     (name: 'bukti', saltype: sql.VarChar(500), value: req.body.bukti ),
137     (name: 'satker_kode_pengisi', saltype: sql.VarChar(20), value: req.body.satker_kode_pengisi ),
138     (name: 'added_by', saltype: sql.VarChar, value: req.body.added_by ),
139     (name: 'keterangan_verifikasi', saltype: sql.VarChar, value: req.body.keterangan_verifikasi )
140  ];
141
142   console.log(req.body.keterangan_verifikasi)
143
144   var query = "UPDATE KINERJA_PENINGKAT_DEV.kinerja_d_037_bukuisbn SET satker_kode@satker_kode,departemen@departemen,id_buku_isbn@id_buku_isbn,judul_buku@judul_buku,tahun
145   executeQuery(res, query, params)
146
147 });
148 router.route('/v_d_077_sertifikasibosen/id').put(function(req, res){
149   var params = [
150     (name: 'id', saltype: sql.Int, value: req.params.id ),
151     (name: 'satker_kode', saltype: sql.VarChar(20), value: req.body.satker_kode ),
152     (name: 'nip_dosen', saltype: sql.VarChar(50), value: req.body.nip_dosen ),
153     (name: 'nama', saltype: sql.VarChar(50), value: req.body.nama ),
154     (name: 'nama_sertifikasi', saltype: sql.VarChar(50), value: req.body.nama_sertifikasi ),
155     (name: 'tema', saltype: sql.VarChar(50), value: req.body.tema ),
156     (name: 'penyelenggara', saltype: sql.VarChar(50), value: req.body.penyelenggara ),
157     (name: 'tanggal_berlaku', saltype: sql.Date, value: req.body.tanggal_berlaku ),
158     (name: 'verifikasi', saltype: sql.Int, value: req.body.verifikasi ),
159     (name: 'bukti', saltype: sql.VarChar(500), value: req.body.bukti ),
160     (name: 'satker_kode_pengisi', saltype: sql.VarChar(20), value: req.body.satker_kode_pengisi ),
```



## 4.4. Query Delete

### 4.4.1. Deskripsi

Membuat query Delete untuk menghapus data.

### 4.4.2. Parameter

Parameter yang digunakan berupa id..

### 4.4.3. Data

Data yang digunakan adalah id dari data yang akan dihapus.

### 4.4.4. Tabel Query

*Tabel 4.161 Detail Query Delete Tabel  
d\_000\_dosen\_inbound*

Route	v_d_000_dosen_inbound/:id
Query	delete from KINERJA_PENDUKUNG_DEV.kinerja.d_000_d osen_inbound where id=@id
Parameter	id
Return	-

*Tabel 4.162 Detail Query Delete Tabel  
d\_000\_dosen\_outbound*

Route	v_d_000_dosen_outbound/:id
Query	delete from KINERJA_PENDUKUNG_DEV.kinerja.d_000_d osen_outbound where id=@id
Parameter	id
Return	-

#### **4.4.4. Source Code**

Query untuk menghapus data di database bisa dilihat di kode 4.4.

```

1 var express = require('express');
2 var router = express.Router();
3 var sql = require('mssql');
4 var executeQuery = require("../Connection");
5 const { query } = require('express');
6
7 var routes = function(){
8   // router.route('/v_d_000_mahasiswa_inbound/:id').delete(function(req, res) {
9   //   var params = [
10   //     { name: 'id', sqltype: sql.Int, value: req.params.id }
11   //   ]
12   //   var query = "delete from KINERJA_PENDUKUNG_DEV.dbo.d_000_mahasiswa_inbound where id=@id"
13   //   executeQuery(res, query, params)
14   // });
15   // router.route('/v_d_000_mahasiswa_outbound/:id').delete(function(req, res) {
16   //   var params = [
17   //     { name: 'id', sqltype: sql.Int, value: req.params.id }
18   //   ]
19   //   var query = "delete from KINERJA_PENDUKUNG_DEV.dbo.d_000_mahasiswa_outbound where id=@id"
20   //   executeQuery(res, query, params)
21   // });
22   router.route('/v_d_000_dosen_inbound/:id').delete(function(req, res) {
23     var params = [
24       { name: 'id', sqltype: sql.Int, value: req.params.id }
25     ]
26     var query = "delete from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_inbound where id=@id"
27     executeQuery(res, query, params)
28   });
29   router.route('/v_d_000_dosen_outbound/:id').delete(function(req, res) {
30     var params = [
31       { name: 'id', sqltype: sql.Int, value: req.params.id }
32     ]
33     var query = "delete from KINERJA_PENDUKUNG_DEV.kinerja.d_000_dosen_outbound where id=@id"
34     executeQuery(res, query, params)
35   });
36   router.route('/v_d_000_publikasi_scopus_scholar/:id').delete(function(req, res) {
37     var params = [
38       { name: 'id', sqltype: sql.BigInt, value: req.params.id }
39     ]
40     var query = "delete from KINERJA_PENDUKUNG_DEV.kinerja.d_000_publikasi_scopus_scholar where id=@id"
41     executeQuery(res, query, params)
42   });
43   router.route('/v_d_037_bukuIsbn/:id').delete(function(req, res) {
44     var params = [
45       { name: 'id', sqltype: sql.BigInt, value: req.params.id }
46     ]
47     var query = "delete from KINERJA_PENDUKUNG_DEV.kinerja.d_037_bukuIsbn where id=@id"
48     executeQuery(res, query, params)
49   });
50   router.route('/v_d_072_sertifikasiDosen/:id').delete(function(req, res) {
51     var params = [
52       { name: 'id', sqltype: sql.Int, value: req.params.id }
53     ]

```

*Kode 4.4 Source Code Query Delete*

## 4.5. Trigger

### 4.5.1. Deskripsi

Dibuat sebuah trigger trigger agar pada saat kolom keterangan\_verifikasi di update oleh user, kolom verified\_date secara otomatis akan diisi dengan tanggal diupdatenya kolom keterangan\_verifikasi.

### 4.5.2. Parameter

Parameter yang digunakan adalah kolom keterangan\_verifikasi.

### 4.5.3. Data

Data yang digunakan adalah waktu dimana form diverifikasi.

### 4.5.4. Source Code

Query untuk menghapus data di database bisa dilihat di kode 4.5.

```
Create Trigger [NAMA_TRIGGER]
on [TARGET_TABLE]
after update
as
if update (keterangan_verifikasi)
begin
    update s
    set verified_date = CURRENT_TIMESTAMP
    from [TARGET_TABLE] as s
    join inserted as new on new.id = s.id
    join deleted as old on old.id = s.id

    where new.keterangan_verifikasi <> ISNULL(old.keterangan_verifikasi,'NULL')
end;
```

*Kode 4.5 Source Code Trigger*

## 4.6. Routing API

### 4.5.1. Deskripsi

Mengarahkan API request Create, Read, Update, Delete pada controller Web API.

### 4.5.2. Parameter

Tidak ada parameter yang digunakan.

### 4.5.3. Data

Data yang digunakan berupa http request dari client.

### 4.5.4. Source Code

Source code untuk routing API bisa dilihat di kode 4.6.

```
var pendukungInsert = require("../Controller/PendukungInsert")();  
var pendukungSelect = require("../Controller/PendukungSelect")();  
var pendukungUpdate = require("../Controller/PendukungUpdate")();  
var pendukungDelete = require("../Controller/PendukungDelete")();  
app.use('/api-dev/pendukungInsert',pendukungInsert);  
app.use('/api-dev/pendukungSelect',pendukungSelect);  
app.use('/api-dev/pendukungUpdate',pendukungUpdate);  
app.use('/api-dev/pendukungDelete',pendukungDelete);
```

*Kode 4.6 Source Code Routing API*

## **BAB V**

### **PENGUJIAN DAN EVALUASI**

Bab ini menjelaskan tahap uji coba dilakukan terhadap fungsi query yang telah kami buat.

#### **5.1 Tujuan Pengujian**

Pengujian dilakukan terhadap fungsi query yang telah dibuat agar fungsi query tersebut telah memenuhi kriteria fungsi yang diminta dan dapat berfungsi sebagaimana mestinya yang dibutuhkan.

#### **5.2 Kriteria Pengujian**

Penilaian atas pencapaian tujuan pengujian didapatkan dengan memperhatikan beberapa hasil yang diharapkan berikut ini:

- a. Kesesuaian dengan kriteria fungsi query yang diminta.
- b. Kesesuaian dengan kriteria fungsi trigger yang diminta.

#### **5.3 Skenario Pengujian**

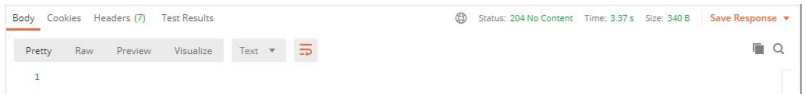
Skenario pengujian dilakukan pada saat query yang sudah selesai dibuat sesuai kebutuhan yang diinginkan akan diuji dengan cara melihat hasil output yang akan ditampilkan menggunakan postman. Jika query sudah berfungsi dengan baik akan diuji dengan front end dari sistem. Fungsi trigger akan diuji melalui DBMS. Untuk routing API secara otomatis akan terbukti berhasil jika query CRUD berhasil dieksekusi.

#### **5.4 Hasil Pengujian**

Terdapat hasil dari code yang telah dibuat pada Bab IV sebelumnya:

### 5.4.1. Query Create

Hasil dari fungsi diatas ada pada gambar 5.1 dan 5.2:



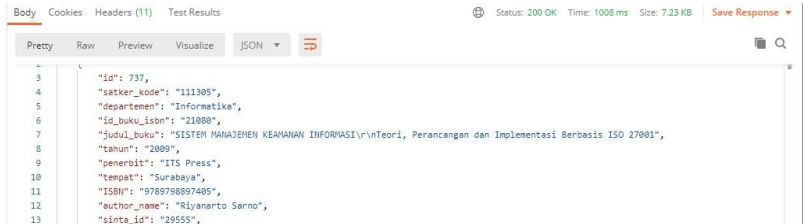
Gambar 5.1 Uji Query Create dengan postman

(037) BUKU ISBN											Home - (037)
Tambah entry											Search: <input type="text"/>
	Departemen	Judul Buku	Tahun	ISBN	Nama Dosen 1	Nama Dosen 2	Nama Dosen 3	Nama Dosen 4	Nama Dosen 5	Verifikasi	AB
181										<input type="checkbox"/>	
182		tes1	2020							<input type="checkbox"/>	
183										<input type="checkbox"/>	

Gambar 5.2 Uji Query Create dengan front end

### 5.4.2. Query Read

Hasil dari fungsi diatas ada pada gambar 5.3 dan 5.4:



Gambar 5.3 Uji Query Read dengan postman

(037) BUKU ISBN											Home - (037) BUKU ISBN
Tambah entry											
	Departemen	Judul Buku	Tahun	ISBN	Nama Dosen 1	Nama Dosen 2	Nama Dosen 3	Nama Dosen 4			
1	Informatika	SISTEM MANAJEMEN KEAMANAN INFORMASI Teori, Perancangan dan Implementasi Berbasis ISO 27001	2009	9789798897405							

Gambar 5.4 Uji Query Read dengan front end

### 5.4.3. Query Update

Hasil dari fungsi diatas ada pada gambar 5.5 dan 5.6:



*Gambar 5.5 Uji Query Update dengan postman*

The screenshot shows the front-end application interface for a book management system. The title is "(037) BUKU ISBN". There is a "Tambah entry" button and a search bar. Below is a table with columns: Departemen, Judul Buku, Tahun, ISBN, Nama Dosen 1, Nama Dosen 2, Nama Dosen 3, Nama Dosen 4, Nama Dosen 5, Verifikasi, AB, and AC. The table contains three rows of data, with the second row highlighted in blue.

	Departemen	Judul Buku	Tahun	ISBN	Nama Dosen 1	Nama Dosen 2	Nama Dosen 3	Nama Dosen 4	Nama Dosen 5	Verifikasi	AB	AC
181										<input type="checkbox"/>		
182		test1	2020							<input type="checkbox"/>		
183	test	test	2020	99999	test					<input type="checkbox"/>		

*Gambar 5.6 Uji Query Update dengan front end*

### 5.4.4. Query Delete

Hasil dari fungsi diatas ada pada gambar 5.7 dan 5.8:



*Gambar 5.7 Query Delete dengan postman*

The screenshot shows the front-end application interface for a book management system. The title is "(037) BUKU ISBN". There is a "Tambah entry" button and a search bar. Below is a table with columns: Departemen, Judul Buku, Tahun, ISBN, Nama Dosen 1, Nama Dosen 2, Nama Dosen 3, Nama Dosen 4, Nama Dosen 5, Verifikasi, AB, and AC. The table contains two rows of data, with the first row highlighted in blue.

	Departemen	Judul Buku	Tahun	ISBN	Nama Dosen 1	Nama Dosen 2	Nama Dosen 3	Nama Dosen 4	Nama Dosen 5	Verifikasi	AB	AC
181										<input type="checkbox"/>		
182		test1	2020							<input type="checkbox"/>		

*Gambar 5.8 Query Delete dengan front end*



#### 5.4.5. Trigger

Hasil dari fungsi diatas ada pada gambar 5.9:

keterangan_verifikasi	verified_date
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
test	2020-10-22 12:53:44.890

*Gambar 5.9 Trigger dengan DBMS*

### 5.5 Evaluasi Pengujian

Hasil evaluasi pengujian fungsi query dapat dilihat pada tabel 5.1.

*Tabel 5.1 Hasil Evaluasi Pengujian Fungsi*

No.	Kriteria Pengujian	Hasil Pengujian
1	Query Create	Terpenuhi
2	Query Read	Terpenuhi
3	Query Update	Terpenuhi
4	Query Delete	Terpenuhi
5	Trigger	Terpenuhi
6	Routing API	Terpenuhi

Dengan hasil pengujian pada tabel di atas, dapat disimpulkan bahwa secara keseluruhan fungsi query telah memenuhi kriteria-kriteria yang sudah disebutkan pada bagian-bagian sebelumnya.

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **6.1 Kesimpulan**

Kesimpulan yang didapat setelah melakukan pembuatan dashboard pada kegiatan kerja praktik di Unit Pengelolaan dan Pengendalian Program (UP3) - Institut Teknologi Sepuluh Nopember Surabaya adalah sebagai berikut:

- Dengan adanya website ini karyawan UP3 ITS dan civitas akademik ITS bisa melakukan input data kinerja dengan efektif.

#### **6.2 Saran**

Saran untuk pengembangan website kinerja adalah sebagai berikut:

- Dilakukan pembaruan data pada database setiap ada perubahan. Agar data bisa selalu ter update

*[Halaman ini sengaja dikosongkan]*

## DAFTAR PUSTAKA

- [1] Guru99, “What is Database What is SQL?,”. [Online] Available  
<https://www.guru99.com/introduction-to-database-sql.html>
- [2] Vuejs, “Introduction,”. [Online] Available  
<https://vuejs.org/v2/guide/>
- [3] Visualstudio, “Getting Started,”. [Online] Available  
<https://code.visualstudio.com/docs>
- [4] Microsoft, “Download,”. [Online] Available  
<https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15>
- [5] Bossanova, “jExcel v3: The javascript spreadsheet,”. [Online] Available  
<https://bossanova.uk/jexcel/v3/>
- [6] Blazemeter, “How to Use Postman to Manage and Execute Your APIs,”. [Online] Available  
<https://www.blazemeter.com/blog/how-use-postman-manage-and-execute-your-apis>
- [7] ITSSOF - ITS Online, “ITS Optimistis Tembus Target Kinerja 2020,” [Online]. Available  
<https://www.its.ac.id/news/2020/02/14/its-optimistis-tembus-target-kinerja-2020/>

*[Halaman ini sengaja dikosongkan]*

## **BIODATA PENULIS**

Nama : Kevin Ashil Faadilah  
Tempat, Tanggal Lahir : Surabaya, 21 Februari 1998  
Jenis Kelamin : Laki-laki  
Agama : Islam  
Status : Belum Menikah  
Alamat Asal : Jalan Rungkut Harapan Blok G-01,  
Surabaya  
Alamat Surabaya : Jalan Rungkut Harapan Blok G-01,  
Surabaya  
Telepon : 082338650500  
Email : richmondroths@gmail.com

## **PENDIDIKAN FORMAL**

2017 – sekarang : Mahasiswa S1 Informatika ITS  
2014 – 2017 : MA PPMI Assalaam Solo  
2010 – 2013 : SMP Negeri 19 Surabaya  
2005 – 2010 : SD Muhammadiyah 4 Pucang  
Surabaya

## **KEMAMPUAN**

- *Web Programming* ( HTML, PHP, CSS, JavaScript)
- *Programming* ( C, C++,C#)
- *Database Manajemen* ( MySQL, SQL Server)
- Sistem Operasi (Windows)
- *Software* Perkantoran (Microsoft Word, Excel, PowerPoint)
- Bahasa (Indonesia, Inggris)

## AKADEMIS

Kuliah : Departemen Informatika, Fakultas Teknologi  
Informasi dan Komunikasi, Institut Teknologi  
Sepuluh Nopember Surabaya

Angkatan : 2017

Semester : 7 (Tujuh)

IPK : 3.11 (Semester 6)



*[Halaman ini sengaja dikosongkan]*